DOCUMENT RESUME

ED 285 584 IR 052 066

AUTHOR Mackenzie, Arch; And Others
TITLE A "Language Lab" for Architectural Design.
PUB DATE Apr 86
NOTE 22p.; Paper presented at the IBM Academic Information Systems University AEP Conference (San Diego, CA, April 5-8, 1986). For the conference proceedings and agenda, see IR 051 770-771. Figures contain very small print.
PUB TYPE Reports - Research/Technical (143) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Animation; *Architectural Character; *Architectural Education; Architectural Programing; Architectural Research; *Building Design; *Computer Assisted Instruction; Computer Graphics; Computer Software; Higher Education; Instructional Effectiveness; *Teaching Methods
IDENTIFIERS *Architectural Grammars Software; *Computer Assisted Design

ABSTRACT

This paper discusses a "language lab" strategy in which traditional studio learning may be supplemented by language lessons using computer graphics techniques to teach architectural grammar, a body of elements and principles that govern the design of buildings belonging to a particular architectural theory or style. Two methods of teaching Frank Lloyd Wright's Usonian grammar to architecture students are described, the manual method and a manual transformation exercise involving the comparison of architectural plans to identify variant and invariant aspects, and it is concluded that both methods yield partial success and partial failure. A computerized synthesis of the two methods, the Architectural Grammars software (AGS), is suggested as a drawing medium. A discussion of AGS describes its use of module-tree templates, set operators, and scripted data structures in five exercises: (1) control-animation, which presents drawing sequences in semi-animated form; (2) variant-animation, which allows a detailed investigation of plan variations; (3) continuous-drawing, which promotes a natural design discourse; (4) continuous-design, which facilitates fast, unself-conscious design decisions; and (5) design-animation, which allows students to reconstruct their own solutions as animations that can be compared directly with the prototype. The text is supplemented with 10 plates. (KM)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED285584

A "LANGUAGE LAB" FOR ARCHITECTURAL DESIGN

Presenter: **Arch Mackenzie** *Cornell University*

copyright © Arch Mackenzie, Cornell, 1986

Coauthors: **Arch Mackenzie** *Cornell University*
Carl Sutter *Cornell University*
Fred Horan *Cornell University*
Scott Hunter *Cornell University*
Chris Kokkinos *Cornell University*
Gail Lidell *Cornell University*

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Arch MacKenzie
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

Much that architecture schools have to teach is embodied in architectural languages. Since Vitruvius codified the elements and principles of classical architecture, architects have written dictionaries to record the evolution of styles. The "linguistic" approach epitomized by the curriculum of the Beaux Arts is reemerging as a focus of design education. Criticism, which is the one–on–one instructional mode of the traditional design studio, is often couched in linguistic terms. Students learn to build "vocabularies," to understand the "syntax" of style and to "read" compositions.

This paper discusses a "language lab" strategy in which traditional studio learning may be supplemented by language lessons using computer graphics techniques. First, it presents a critique of a manual version of the Usonian Exercise that was designed to teach the grammar of Frank Lloyd Wright's residential architecture. Then it describes AGS, the Architectural Grammars software, that we have developed to run on IBM PC/ATs, and presents the design–animation and continuous–drawing exercises, using AGS, that improve on the manual methods.

The paper outlines the technique of design–animation, which is achieved by a special application of the AGS "module tree" editor and which demonstrates how building design variations are generated by the rules of architectural grammars; and the technique of continuous–designing, which is implemented by using the AGS set operators, instead of lines, to make drawings and which complements the analytical posture of the animations by giving students practice in rapid, unself–conscious designing. It also shows how the AGS scripted data

IR052066

structure is used, for purposes of critique, to record the stream of design operations and to play back the script for review.

The AGS techniques and the lesson exercises are designed to promote both grammatical competence and intuition for designing within particular architectural styles. Presented here as part of a strategy for teaching architectural languages, these didactic tools are compatible with the theory that design is a reciprocal process between critique and creation.

THE USONIAN LANGUAGE LESSON

ARCHITECTURAL GRAMMARS

An architectural grammar is a body of elements and principles that governs the design of buildings belonging to a particular architectural theory or style. Young architects often swerve from their masters toward an architecture of their own. To swerve, however, one must first be literate, as Frank Lloyd Wright became during this apprenticeship to Sullivan. Wright, in turn, developed his own implicit grammatical system for the Prairie Houses and then swerved from this work to produce the Usonian Houses. I have researched the latter, made its grammar explicit and used it to demonstrate the relation between stylistic literacy and the process of invention. Presented below is a brief description and critique of two manual techniques I have used to teach the Usonian grammar and a design specification for the AGS (Architectural Grammars) software that results from the critique.

ELEMENTS OF THE USONIAN GRAMMAR

The objective of the first Usonian design exercise is to teach the rules of Wright's residential architecture by asking students to design a house on a unique site using the Usonian grammar. This is a simple two stage process in which students first review Wright's rules, find examples of their application in his work and commit them to memory. In the second phase, they are given a site plan and a space requirements program and asked to design a grammatically correct Wrightian house. An important variation on this exercise is to add a third step in which students design a second house that swerves from the orthodoxy of the grammar.

At the beginning of the exercise, students are given the complete grammar, including a catalogue of rules and an explanation of their derivation from Wright's first principles. Plates 1 and 2 are representative sheets from the

catalogue and display some of the rules that were codified and organized into "domains," as exemplified by the title "House Exterior" on sheet one and "House Interior" on sheet two. Each rule is represented by an "icon" and most of the rules include direct quotations of Wright's aphorisms. In the first rule on plate 1, for example, (ignore the header, which is a sorting code and check list for the author) Wright says, "the proper scale for a building is the human being...bring the house down to fit the normal man...broaden out the mass...keep the roof line low." or , in the first rule on Plat 2, "it is the shape of shelter that really gives a sense of the outside coming in or the inside going out...release...interior space to the outside." In each case, the rule, qualified by diagrams, defines a configuration or an operation to be performed in the design of these houses.

STUDENT DESIGN PROTOCOL

Plates 3a, b, c and d illustrate a design protocol that documents one student's efforts. In this case, the grammar was given to a class of second year architecture students, who did not know whose architecture it represented and who were given three days, with no criticism, to complete the exercise. The protocol, which was compiled by the student from his drawings, shows sixteen design steps, four steps on each plate, which are numbered on the lower right of each set of sketch drawings. One can just barely make out the grammar icons that the student used to annotate his design process, a good example being step seven, where many icons appear at the left of the sketches. The student's work reaches a level of detail that makes a comparison with Wright's floor plan instructive. Note particularly, the similarity between the last drawing in the protocol (lower right, plate 3d) and the ell-shaped floor plans of Wright's houses in bottom row of plate 4. The student design and Wright's houses have all the same characteristics, a result to be expected if the grammar were complete and correctly applied. These results demonstrate the predictive power of the grammar, and as such, suggest that it is a comprehensive theory of Wright's small houses.

CRITIQUE OF THE EXERCISE

The manual exercise appeared to be successful. The student was able in a short period, without criticism and without prior knowledge of the architect, to design a house using the grammar that was remarkably like the prototype ell-shaped house. In the first stages of design he carefully considered alternative solutions to parts of the problem (steps one through seven) and then, after having become

convinced of the outline of his design, proceeded to develop the plan in some detail. Even in this exemplary project, however, good working method that explores the latitude of the grammar gave way to multiple trials for solving minor problems with proportion and detail (steps 8–14). The generative capability of the grammar is obscured by the student's preoccupation with solving immediate design problems.

TRANSFORMATION STUDIES

Transformation studies, which are an alternative method for learning the grammar, demonstrate how it generates unique designs. The charts in plates 4 and 5 display six floor plans, representing the range of orthogonal houses that may be generated by the Usonian grammar. In a didactic version of the study, some of the plans would be prepared by the instructor as control conventions and some by students as variations relative to the control.

The first three plans are ell–shape, or "polywog" types, while the last three are tee–shape, two–story and "in–line" types, respectively. The transformation, beginning at the top of each column and working downward, shows the general to specific build–up of information that governs the design of the completed house plan at the bottom. The transitions at each step may then be compared with those of the other house diagrams in the same row. Finally, the operations required to transform each diagrammatic stage into the next are isolated, translated into operational rules and correlated with comparable rules stated implicitly in Wright's aphorisms about the principles of the "natural house" (plates 1 and 2). By comparing the diagrams in each row from the top–down, one can easily identify the variant and invariant aspects of the plans. The ell–shaped houses, of course, share many common attributes, and they vary in limited and predictable ways. The other houses differ significantly from the ell–shapes and from each other, but one can detect without difficulty the general similarities of style linking all the house.

CRITIQUE OF THE TRANSFORMATION EXERCISE

While the manual transformation exercise is useful to illustrate the variant and invariant aspects of the grammar and its generative capability, its drawbacks stem from the limitations of traditional drawing method. Using this medium, students inevitably become preoccupied with method rather than content; and, while the manual method of constructing transformations explains how gram-

matically correct solutions are produced, it does not help designers develop a grammatical intuition.

CAI SOFTWARE TO IMPROVE THE USONIAN PROTOTYPE LANGUAGE LESSON

The two versions of the Usonian language lesson discussed above have been employed with partial success and partial failure. The traditional design problem format draws students into a familiar problem–solving process and yet it also encourages them to focus on the search for a specific solution without comprehending the generality of the underlying grammar. Moreover, some students, who understand the operational rules as constraints rather than as generators, become inhibited in design. Contrarily, the transformation studies make the generative power of the grammar explicit, but they require tedious, repetitive drawing and thus cause students to focus more on method than on content. These experiences make it clear that a synthesis of these approaches is required and that a new medium for implementing them must be found. The following describes the elements of such a synthesis and of the Architectural Grammars software that was created to serve as the drawing medium.

LANGUAGE LESSONS USING THE ARCHITECTURAL GRAMMAR SOFTWARE

CONTROL–ANIMATION EXERCISE

There are five lessons in the AGS version of the Usonian language class: control–animation, variant–animation, continuous–drawing, continuous–design and design–animation. Control–animation on AGS is purely explanatory. Its function is to present the same type of drawing sequence as the transformation but in a semi–animated form, which the designer may review a step–at–a–time. Plate 6 enters an animation of the Pope–Leighey house plan in the sixth step. The diagram on the left is the operand (step five), the symbol highlighted in the chart in the middle is the relevant operator, or rule, from the Usonian grammar and the diagram on the right is the result of the operation (step six). The left hand diagram was the product of the previous operator "place the quadrants" of the house plan, which includes the bedroom, work area, (kitchen), living room and carport, divides the plan and modifies and base configuration to fit the constraints of site and program. "This figure was then operated on to produce the right–hand diagram by the operator, "align the elements," which

produces a grid for positioning such masonry elements, as the fireplace and living room pier.

Whereas the drawing time required to construct the manual version of the transformation will always limit the number of steps that can be displayed, the animated AGS version can easily accommodate multiple step transformations. Semi-animation, which allows the instructor to isolate steps, is a dramatic way to present the effect of the grammatical operators; and one understands intuitively from this demonstration the distinction between the variant and invariant rules of a grammar. Instant recognition of these concepts is a useful hook both to simulate students' curiosity and to show them the utility of this analytical method.

VARIANT ANIMATION EXERCISE

Plate 7 illustrates four steps in the variant animation exercise. In this case students would be asked to create an animation of the Jacobs House (plate 4, column 1) to run simultaneously with the control-animation. The objective is to engage students in a detailed investigation of plan variations. As the variant and control plans vary in the example, one can see that the first steps are identical. But when the living and carport areas of the respective plans are generated, the control plan places the carport opposite the living area, making a cross-shaped diagram. The variant plan, on the other hand, places the entry perpendicular to the living area, which requires the carport to abut the central core and the main entry to slide into the living area along the side of the core. From this point forward, similar operators produce different results in the floor plans.

The Jacobs House animation illustrates a single variation that shows explicitly how the grammar produces similar plans. The animations for the Pew and Goetsch-Winkler Houses (plate 5, row 4, house plans 5 and 6), are more advanced variations on the conventional ell-shaped configuration. They require different planning grids, proportions and masonry elements than do the ell-shaped prototypes. To model these variations, students would have to reconfigure these elements. How well they match these variations against the control will be immediately discernible to the instructor who can step back through the animation.

MODULE–TREE TEMPLATES

AGS can be used to build trees of "form–modules," or sets of polygons or lines that have been created as modules by the designer. This function was used to build the tree in plate 8, which is the skeleton for the original control–animation. The template was also used to structure the variant–animation. Thus to create an animation exercise an instructor would first build a tree as a module "outline" for the control and variant plans. Students would then use the tree as a template to construct the variations assigned to them.

By toggling "focus" on the module line, for example, one can display particular branches of the tree. Plate 9 is a case that shows steps five and six of the animation. Notice that in the tree for each step the instructor created modules named "rgt_#, lft_# and cen_#" to identify those drawing modules that are appropriate to the right, left and center drawings in the animation. Below these designations are drawn the overlays for each step. To modify the leftmost drawing in the animation in order to represent a step in a new plan variation, students need only to use the graphics editor to delete the module names for the existing forms, to draw a replacement for it and to link the new module to the tree.

CONTINUOUS DRAWING EXERCISE

Whereas the animation studies are comparable to language laboratory exercises that teach normative and variant grammatical rules, the continuous drawing exercise is analogous to conversation class. The objective of the animation studies is to help students build a rational model of a particular grammatical structure. The objective of the continuous drawing exercise is to help them develop a grammatical intuition. As in conversation, the unself–conscious design process proceeds by meaningful transactions. Conversants find themselves in a specific context and this leads to unpremeditated discourse. In the original Usonian Exercise, students were often initially inhibited by the rules. They became self–conscious about each move. Instead, continuous– drawing is intended to promote a natural design discourse that occurs so rapidly that students will not have time to be self–conscious.

SET OPERATORS

AGS employs set operators instead of lines and erasures as the medium for continuous drawing. They allow designers to add, subtract, multiply or divide

shapes directly in ways consistent with the formal geometries inherent in "legible" building designs. At the left of plate 10 is a list of set operators employed by AGS. Notice particularly the column called simple polygons, which displays on the left of each box two intersecting rectangles. The leftmost box is the operand, the rightmost is the operator and, at the right of each box, is the transform. The result of an add operation, for example, would be the elimination of the intersection of the rectangles, while for the subtract operation, the corner of the operand is cut-out. In AGS, continuous drawing proceeds by utilizing the operators at the bottom of the matrix, enabling the design to place, combine, add, subtract, divide, clip, cut-out, replicate, rotate, reflect, increase or decrease drawings without need of erasures or revisions.

SCRIPTED DATA STRUCTURE

In the continuous drawing exercises students would be asked to redraw one of the Usonian houses without stopping to erase. AGS has a "scripted" data structure that would record the entire sequence of operations. The script can be played-back to isolate the drawing operations used to reconstruct the plan of the House. When a particular transaction is isolated, the operators and drawing primitives used in executing it are written onto the prompt line. This gives both students and their instructors the chance to review the continuous drawing process.

CONTINUOUS-DESIGN EXERCISE

Animation and continuous drawing teach grammar through the replication of existing designs. Fluency in an architectural language, however, means being able to design novel plans. To teach fluency, the order of these exercises is reversed and they are recast in an active design mode.

In the continuous-design exercise, students would be given unique sitting and programatic conditions and asked to design a house using Wright's grammar. Like continuous-drawing, the intention of continuous-designing is to get students to make fast, unself-conscious design decisions without stopping to make erasures. The procedure may be repeated several times in order to encourage students to respond without hesitation to the circumstances of the problem setting.

The students' outward stance toward the process are likely to be symptomatic of their state-of-mind. If they have emerged from the previous exercises feeling

competent and having an intuitive sense of what the grammar is all about, they may move confidently toward unique solutions. If they do not have a grasp of the grammar nor a feeling for how far it may be stretched, then they may proceed too self-consciously. In the later case, students' first trials may be highly conventionalized or they may have difficulty in completing the exercise. The AGS scripting capability would let an instructor intervene at this point to review a student's design script with him and to suggest other approaches. If students still have trouble after the lesson set is complete, however, they would be able to go back and redo the exercise.

DESIGN-ANIMATION EXERCISE

If the continuous-design exercise would stretch students' intuitive grasp of the grammar, the design-animation exercise would reinforce their rational understanding of its generative logic by having them reconstruct their own solutions as animations that can be compared directly with the prototype. In this way, they could test the fidelity of their designs to the grammatical boundaries of the original architectural style. By this point in the language lesson, students would have examined enough variations to understand the integrity of the grammar and should be able to judge their own work in relation to it. If their designs could be easily reconstructed using the animation template, it would indicate that they are derivative. If they do not easily fit, then it would suggest that, in some significant way, they are not a product of the grammar. In the former case, students might consider whether their solutions are normative or innovative. In the latter, whether they are deviant or evolutionary.

LANGUAGE LESSONS AND THE DESIGN PROCESS

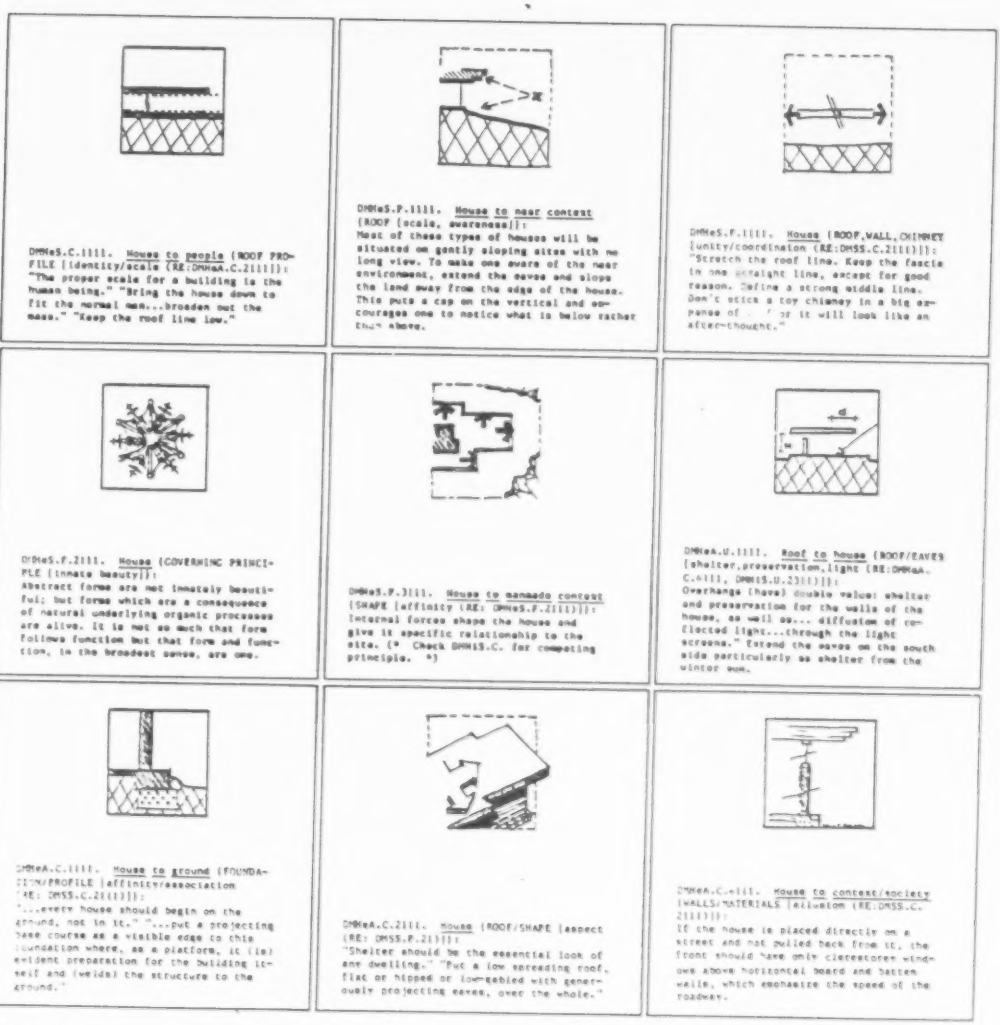
The language lesson exercises described above, which could not be done without the key computer graphic features of AGS, represent an improvement over the techniques of trial and error design and over the manually constructed transformations. The module tree, the scripted data structure and the set operators, respectively, will give instructors a greater degree of control in preparing and critiquing design exercises and will engage students more naturally and more consciously in learning architectural language.

Animations using AGS make the generative power and latitude of grammars explicit, and templates enable instructors to design the control situations and make it easy for students to construct variations. In a general way, animations are a device to draw students into the notion that architecture is not a history of

individual buildings alone, but one of styles; and that to make innovative architecture, one must first understand and be conversant with architectural languages.

Similarly, the continuous–drawing and design exercises would be difficult to do without the AGS set operators. Because the operators are a fast and logical drawing medium, they facilitate unself–conscious designing that simultaneously lays down an implicit rationale. The scripted data structures record this sequence of operations, allowing designers to review their unpremeditated processes for clues to unravel this rationale. The respective roles of continuous–design and design–scripting in the language lesson, as conversation and critique, complete the circle of intellectual inquiry and intuitive exploration necessary to link the academic lesson with the practice of design.

HOUSE EXTERIOR



DMHeS.C.1111. House to people (ROOF PROFILE [identity/scale (RE:DMHeA.C.2111)]:
"The proper scale for a building is the human being." "Bring the house down to fit the normal man...broaden out the mass." "Keep the roof line low."

DMHeS.P.1111. House to near context (ROOF [scale, awareness]]:
Most of these types of houses will be situated on gently sloping sites with no long view. To make one aware of the near environment, extend the eaves and slope the land away from the edge of the house. This puts a cap on the vertical and encourages one to notice what is below rather than above.

DMHeS.F.1111. House [ROOF,WALL,CHIMNEY [unity/coordinaion (RE:DMSS.C.2111)]]:
"Stretch the roof line. Keep the fascia in one straight line, except for good reason. Define a strong middle line. Don't stick a toy chimney in a big expanse of ... / or it will look like an after-thought."

DMHeS.F.2111. House (GOVERNING PRINCIPLE [innate beauty]]:
Abstract forms are not innately beautiful; but forms which are a consequence of natural underlying organic processes are alive. It is not so much that form follows function but that form and function, in the broadest sense, are one.

DMHeS.P.3111. House to nomadic context (SHAPE [affinity (RE: DMHeS.F.2111)]]:
Internal forces shape the house and give it specific relationship to the site. (* Check DMHiS.C. for competing principle. *)

DMHeA.U.1111. Roof to house (ROOF/EAVES [shelter,preservation,light (RE:DMHeA.C.4111, DMHiS.U.2311)]]:
Overhangs (have) double value: shelter and preservation for the walls of the house, as well as... diffusion of reflected light...through the light screens." Extend the eaves on the south side particularly as shelter from the winter sun.

DMHeA.C.1111. House to ground (FOUNDATION/PROFILE [affinity/association (RE: DMSS.C.2111)]]:
"...every house should begin on the ground, not in it." "...put a projecting base course as a visible edge to the foundation where, as a platform, it (is) evident preparation for the building itself and [welds] the structure to the ground."

DMHeA.C.2111. House (ROOF/SHAPE [aspect (RE: DMSS.P.2111)]]:
"Shelter should be the essential look of one dwelling." "Put a low spreading roof, flat or hipped or low-gabled with generously projecting eaves, over the whole."

DMHeA.C.-111. House to context/society [WALLS/MATERIALS [allusion (RE:DMSS.C.-2111)]]:
If the house is placed directly on a street and not pulled back from it, the front should have only clerestories windows above horizontal board and batten walls, which emphasize the speed of the roadway.

Plate 1.

BEST COPY AVAILABLE

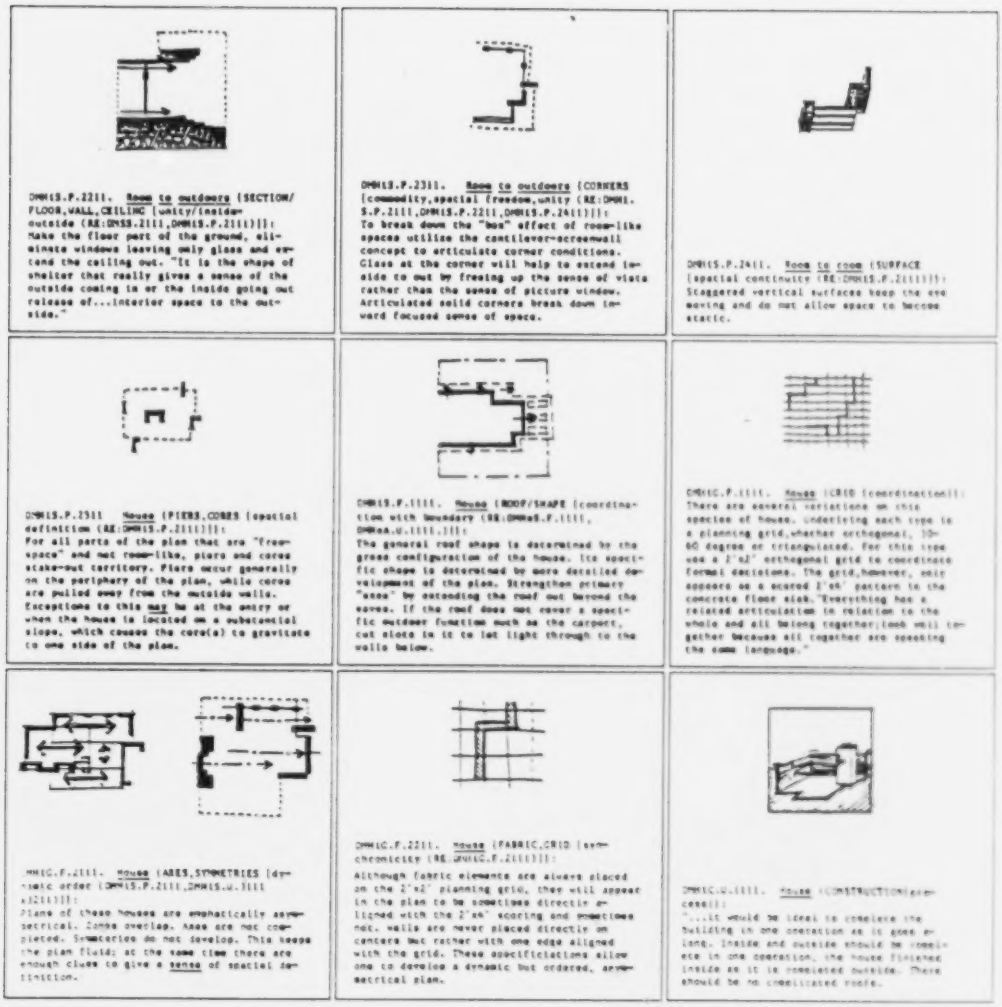


Plate 2.

BEST COPY AVAILABLE

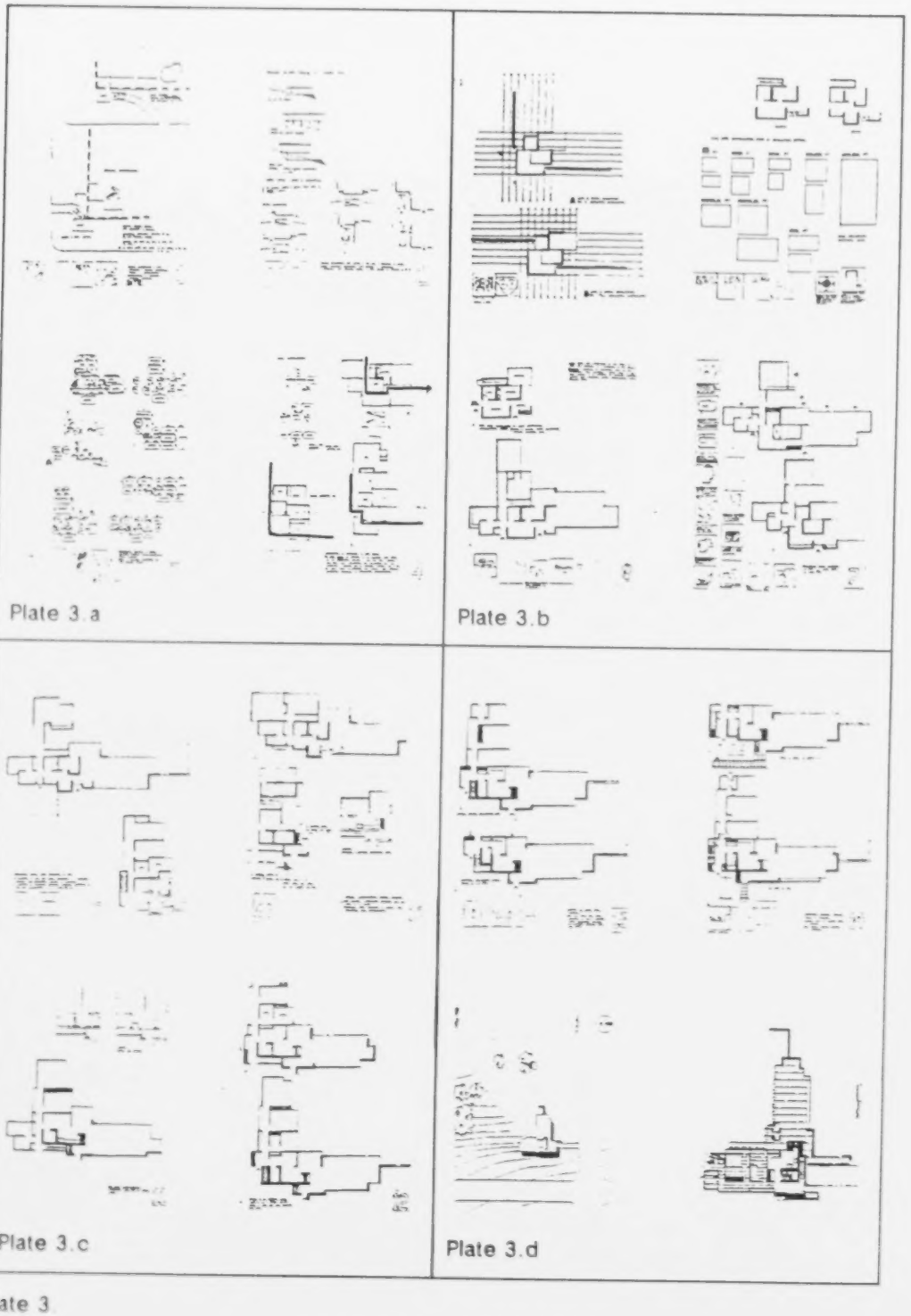


Plate 3.a

Plate 3.b

Plate 3.c

Plate 3.d

Plate 3

BEST COPY AVAILABLE

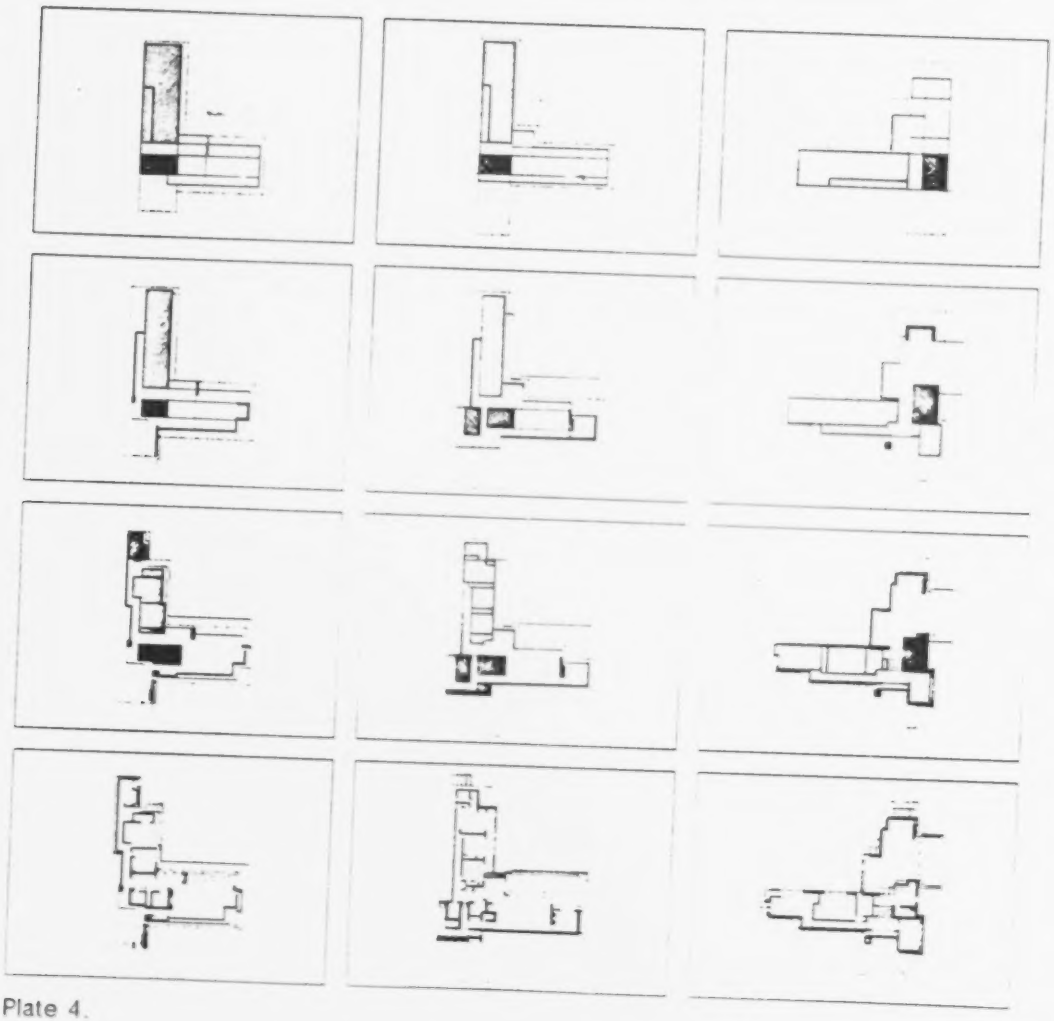


Plate 4.

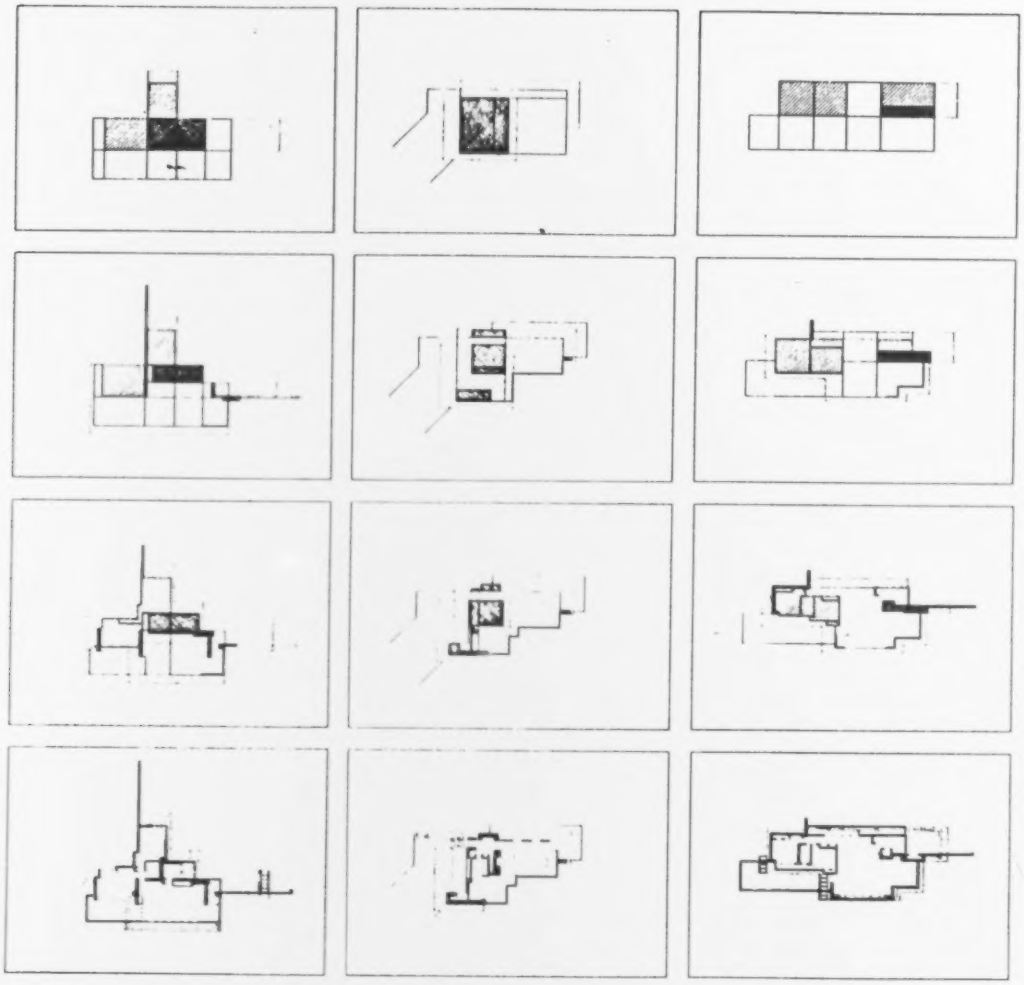


Plate 5.

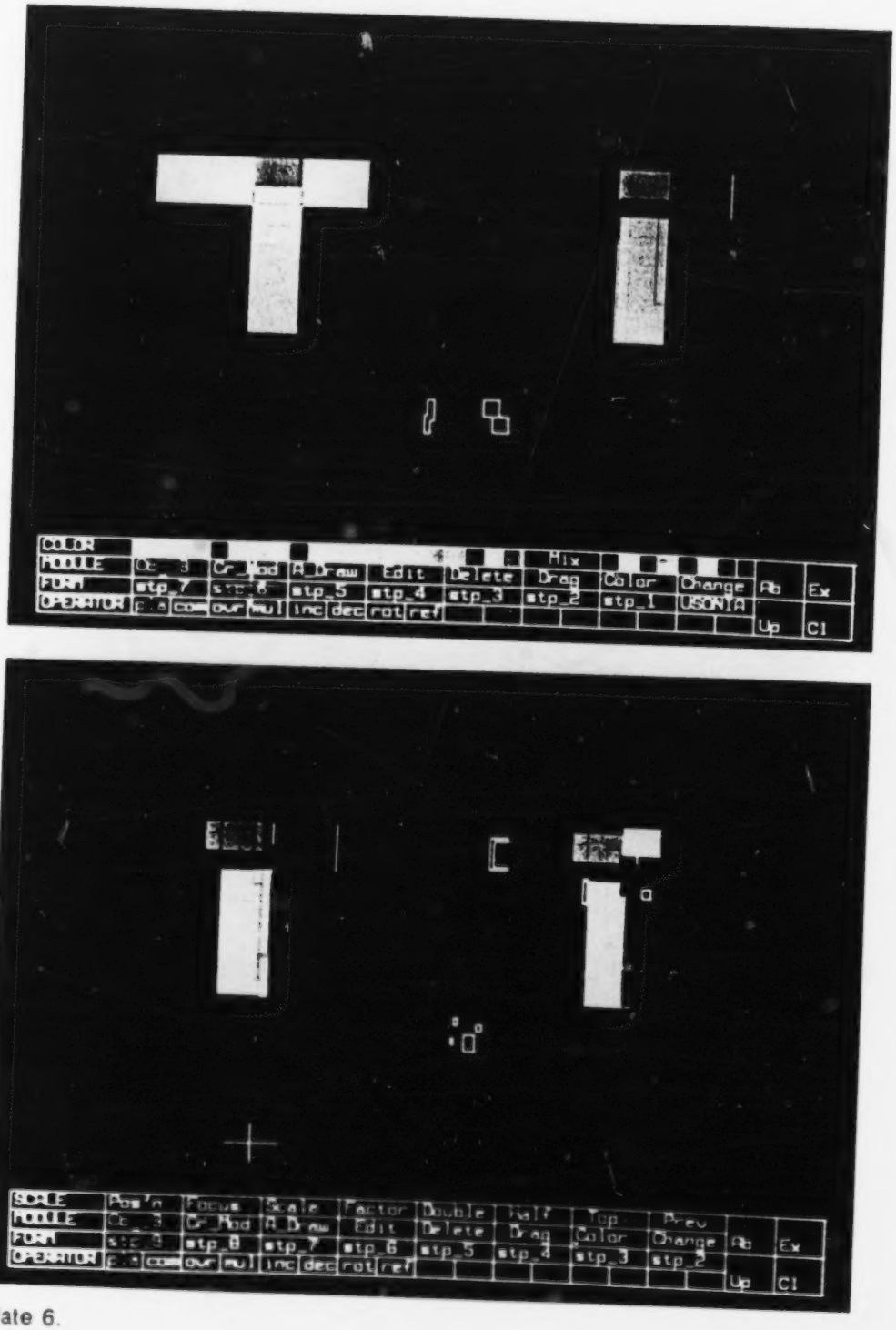

COLOR
MODULE
FORM
OPERATION
Mix
Edit
Delete
Drag
Color
Change
USONIA
Ab
Ex
Up
Cl
over
mul
inc
dec
rot
ref
SCALE
Focus
Scale
Factor
Double
Half
Top
Prev

Plate 6.

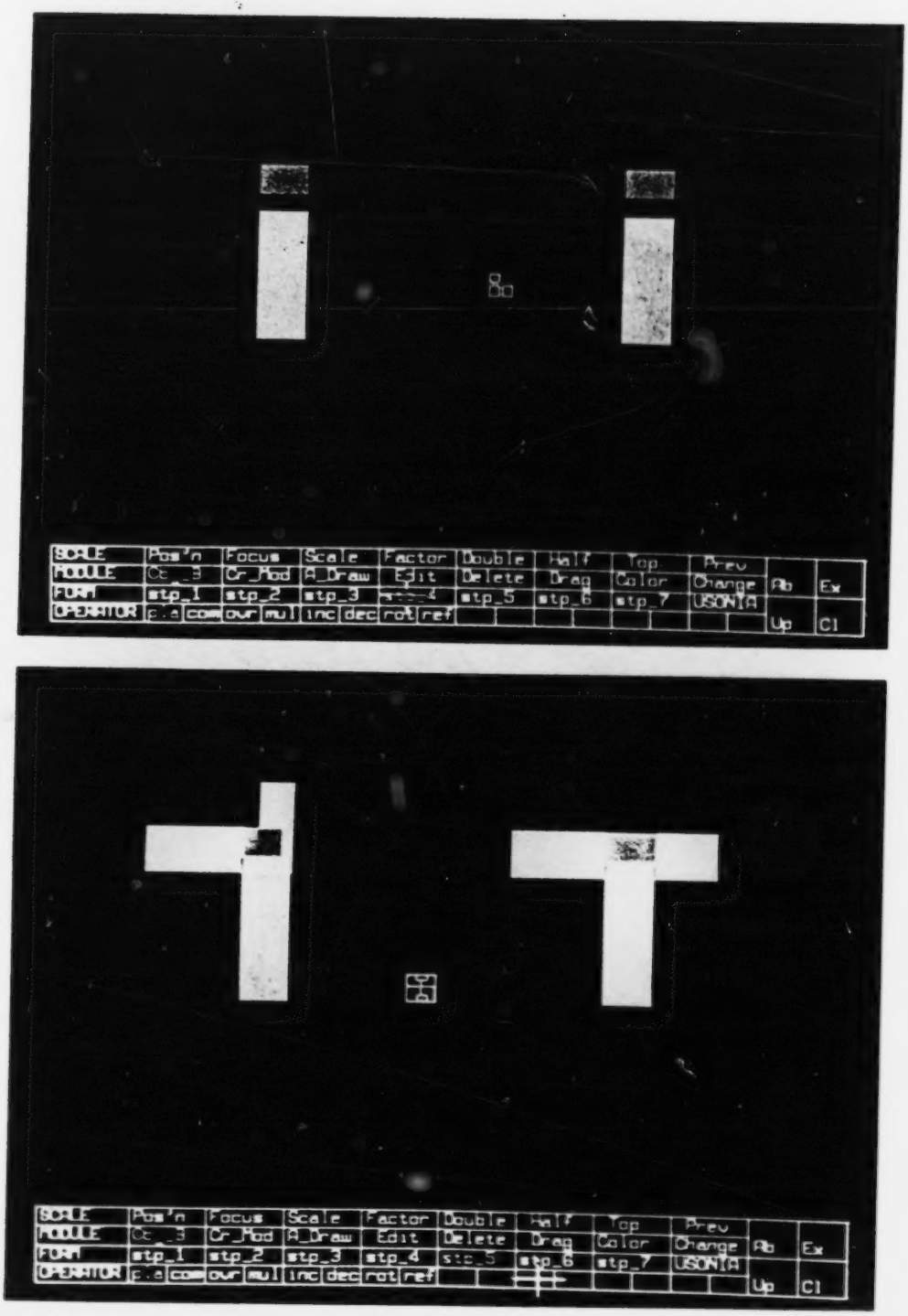

SCALE Pos'n Focus Scale Factor Double Half Top Prev
MODULE Cr_Mod A_Draw Edit Delete Drag Color Change Ab Ex
FORM stp_1 stp_2 stp_3 stp_4 stp_5 stp_6 stp_7 USONIA
OPERATION p.e com ovr mul inc dec rot ref Up Cl

Plate 7.a

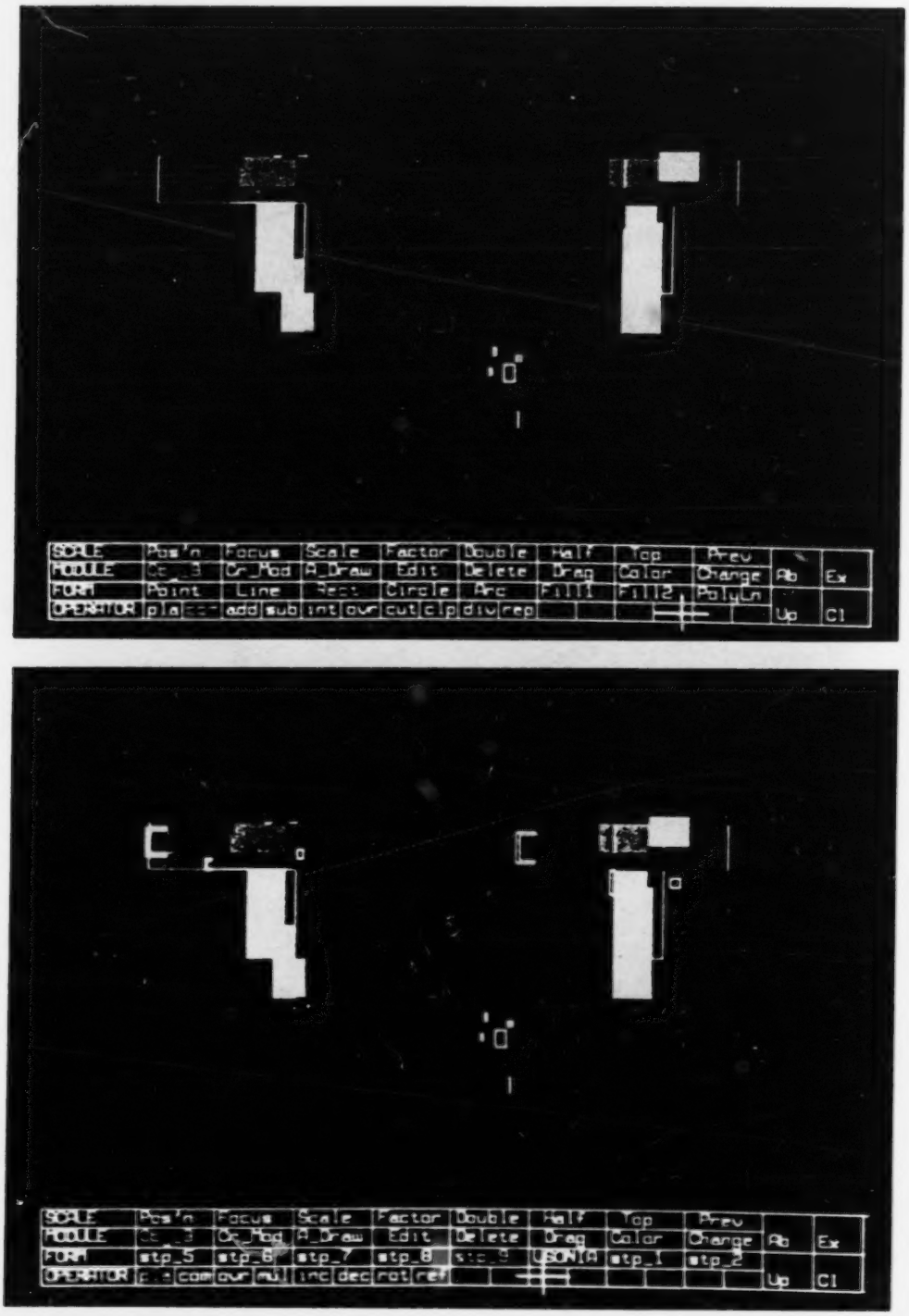

SCALE Pos'n Focus Scale Factor Double Half Top Prev
MODULE Cr_Mod A_Draw Edit Delete Drag Color Change Ab Ex
FORM Point Line Rect Circle Arc Fill1 Fill2 PolyLn
OPERATOR add sub int ovr cut clp div rep Up Cl
SCALE Pos'n Focus Scale Factor Double Half Top Prev
MODULE Cr_Mod A_Draw Edit Delete Drag Color Change Ab Ex
FORM stp_5 stp_6 stp_7 stp_8 stp_1 stp_2
OPERATOR com ovr mul inc dec rot ref Up Cl

Plate 7.b

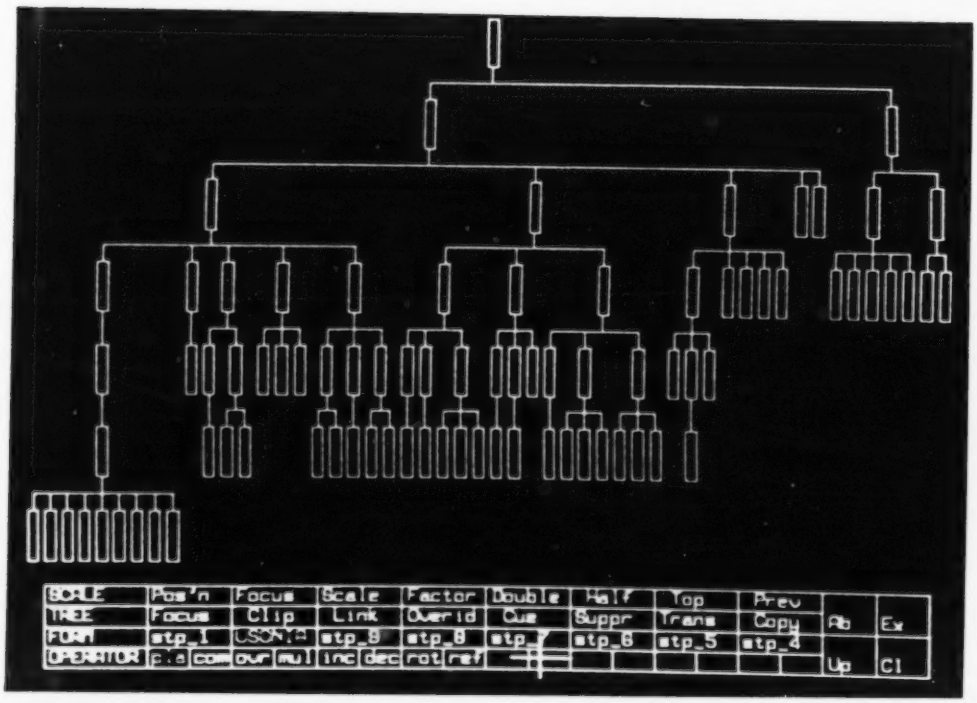
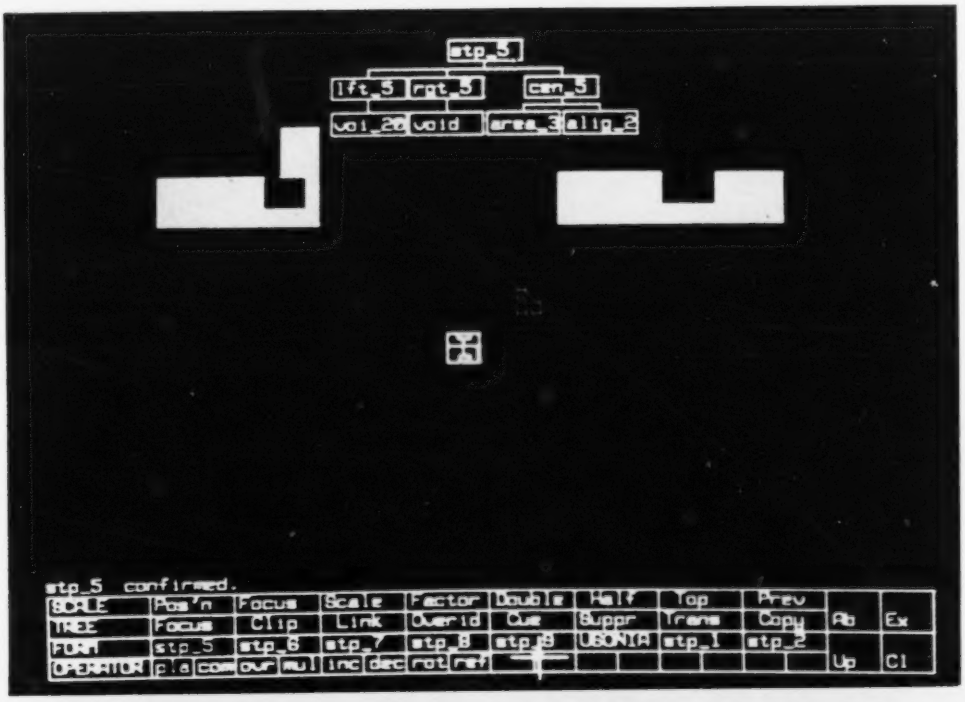

SCALE
Pos'n
Focus
Scale
Factor
Double
Half
Top
Prev
TREE
Focus
Clip
Link
Overid
Cue
Suppr
Trans
Copy
Ab
Ex
FORM
stp_1
USONYC
stp_9
stp_8
stp_7
stp_6
stp_5
stp_4
OPERATOR
pick
com
ovr
mul
inc
dec
rot
ref
Up
Cl

Plate 8.


stp_5
lft_5
rgt_5
cen_5
voi_20
void
area_3
alig_2
stp_5 confirmed.
SCALE
TREE
FORM
OPERATION
Pos'n
Focus
Clip
stp_5
stp_6
Scale
Link
stp_7
Factor
Overld
stp_8
Double
Cue
stp_8
Half
Suppr
USONIA
Top
Trans
stp_1
Prev
Copy
stp_2
Ab
Up
Ex
Cl
p:a
com
cur
mul
inc
dec
rot
ref

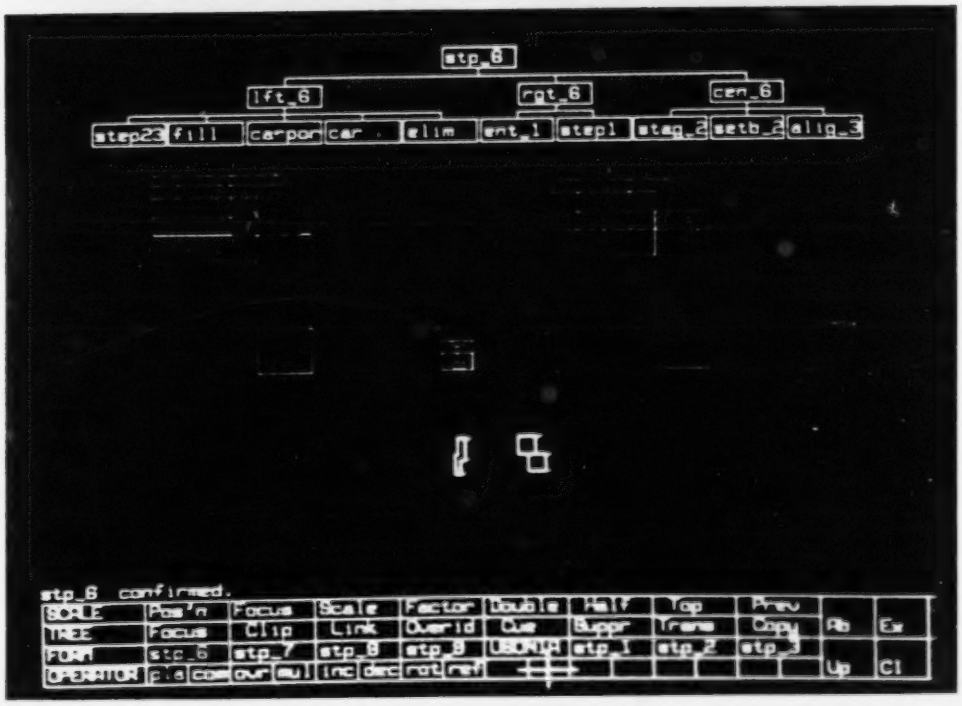

stp_8
lft_6
rgt_6
cen_6
step23
fill
carpor
car .
elim
ent_1
step1
steq_2
setb_2
alig_3
stp_8 confirmed.
SCALE
TREE
FORM
OPERATION
Pos'n
Focus
Clip
stp_6
stp_7
Scale
Link
stp_8
Factor
Overld
stp_8
Double
Cue
stp_8
Half
Suppr
USONIA
Top
Trans
stp_1
Prev
Copy
stp_2
stp_3
Ab
Up
Ex
Cl
p:a
com
cur
mul
inc
dec
rot
ref

Plate 9

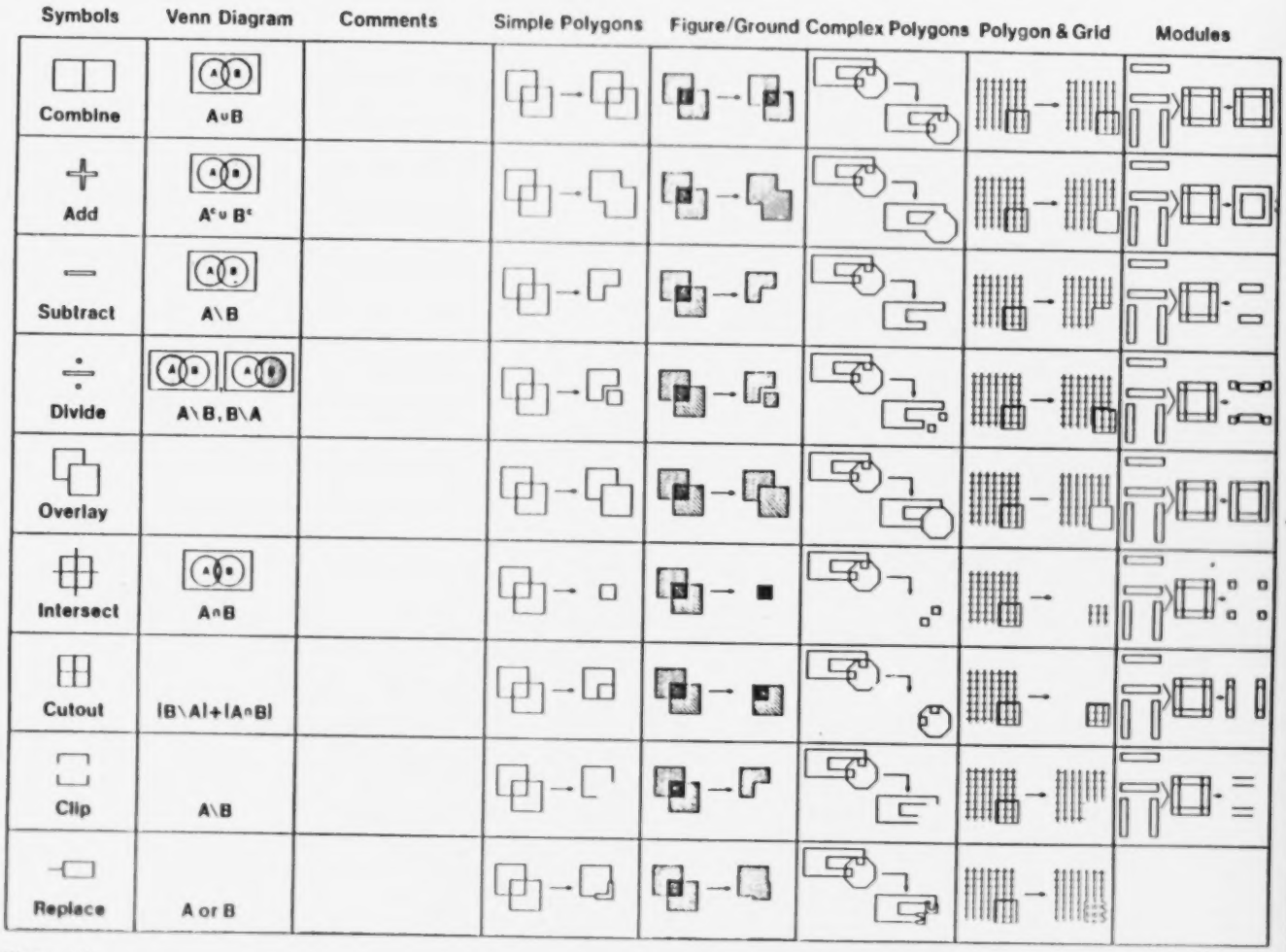

Symbols
Venn Diagram
Comments
Simple Polygons
Figure/Ground
Complex Polygons
Polygon & Grid
Modules
Combine
A∪B
Add
A^c ∪ B^c
Subtract
A\B
Divide
A\B, B\A
Overlay
Intersect
A∩B
Cutout
|B\A|+|A∩B|
Clip
A\B
Replace
A or B

Plate 10.